UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/533,042	03/22/2000	Ryuichi Sunayama	826.1597/JDH	7804

21171 7590 04/20/2004

STAAS & HALSEY LLP
SUITE 700
1201 NEW YORK AVENUE, N.W.
WASHINGTON, DC 20005

EXAMINER	
KNAPP, JUSTIN R	

ART UNIT	PAPER NUMBER
2182	11

DATE MAILED: 04/20/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 09/533,042	Applicant(s) SUNAYAMA ET AL.	
	Examiner Justin Knapp	Art Unit 2182	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *30 January 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *2,4 and 6-19* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *2,4,6-19* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Papers Submitted

1. It is hereby acknowledged that the following papers have been received and placed of record in the file: Extension of time as received 1/30/04.

Claim Objections

2. Claims 2, 4, 6, and 14-18 are objected to because of the following informalities: a) each claim states, "wherein the predicted branch is the branch of the instruction equivalent to the subroutine return in an architecture for which a particular instruction for a subroutine is not prepared". Examiner believes the word, -- return -- should be inserted between "subroutine" and "is not prepared". Appropriate correction is required.

Claim Rejections - 35 USC § 112

3. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

4. Claims 4 and 14-18 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. The claim language, "and wherein only the taken instruction of a branch instruction is thus registered" is indefinite as far as how it relates to any other part of the claim. The Office believes that the claim language was supposed to read "and wherein only a taken branch instruction is thus registered". If so, it would put claims 4 and 14-18 in condition for allowance.

Claim Rejections - 35 USC § 102

5. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

6. Claims ~~1-18~~ 2,4-6,19 are rejected under 35 U.S.C. 102(b) as being anticipated by Losq, "Subroutine Return Address Stack", IBM Technical Disclosure Bulletin, December 1981.

7. Referring to claims 2 and 16, Losq has taught:

a) a storing circuit storing information specifying a return address of a subroutine when an instruction equivalent to a subroutine call is detected. On page 1, lines 26-27 and page 2, lines 1-8, Losq has taught a storing circuit consisting of a Branch History Table and Subroutine Return Address Stack storing information specifying a return address of a subroutine when a call is detected.

b) a comparing circuit making a comparison between information specifying a branch destination address of an instruction which can possibly be an instruction equivalent to a subroutine return and the information specifying the return address stored in said storing circuit, and outputting a result of the comparison, when the instruction which can possibly be the instruction equivalent to the subroutine return is detected. On page 2, lines 40-46, Losq has taught a decoder used as comparing circuitry to compare R1 of a BALR or similar instruction on the return stack, which

holds information specifying the return address, and R2 of a BCR or similar instruction which holds the branch destination address of a potential subroutine return instruction.

c) an identifying circuit identifying an instruction equivalent to a subroutine return, which corresponds to the instruction equivalent to the subroutine call, based on the result of the comparison. It is inherent in the system disclosed by Losq, that identifying circuitry must be present since Losq's system is able to identify an instruction equivalent to a subroutine return.

d) wherein the predicted branch is the branch of the instruction equivalent to the subroutine return in an architecture for which a particular instruction for a subroutine is not prepared (see page 2, lines 22-32), wherein said storing circuit which stores a register number of a link register, which is specified by the instruction equivalent to the subroutine call, as the information specifying the return address (it is inherent the return address stack (storing circuit) stores a register number of a link register specified by a BALR instruction (potentially an instruction equivalent to a subroutine call) due to the format of a BALR instruction (see IBM ESA/390 Principles of Operation, pages 7-14 – 7-15) and wherein said storing circuit stores the return address of the subroutine as the information specifying the return address (See page 1, lines 24-27 and page 2, lines 1-13).

8. Referring to claims 4 and 17, Losq has taught:

a) a stack circuit storing information specifying a return address of a subroutine (See page 1, lines 25-27 and page 2, lines 1-4).

b) a push circuit pushing the information specifying the return address onto said stack circuit, when an instruction equivalent to a subroutine call is detected (Page 1, lines 26-27 and page 2,

lines 1-2 inherently show push circuitry exists to push information specifying a return address down onto the stack).

c) a comparing circuit making a comparison between information specifying a branch destination address of an instruction which can possibly be an instruction equivalent to a subroutine return and the information specifying the return address stored in a top entry of said stack circuit, and outputting a result of a comparison, when the instruction which can possibly be the instruction equivalent to the subroutine return is detected. As explained herein above, this is inherent.

d) an identifying circuit identifying an instruction equivalent to a subroutine return, which corresponds to the instruction equivalent to the subroutine call, based on the result of the comparison. As explained herein above, this is inherent.

e) wherein the predicted branch is the branch of the instruction equivalent to the subroutine return in an architecture for which a particular instruction for a subroutine is not prepared, and wherein only the taken instruction of a branch instruction is thus registered. (see page 2, lines 22-32).

9. Referring to claim 6, Losq has taught

a) a stack circuit storing information specifying a return address of a subroutine (See page 1, lines 25-27 and page 2, lines 1-4).

b) a push circuit pushing the information specifying the return address onto said stack circuit, when an instruction equivalent to a subroutine call is detected (Page 1, lines 26-27 and page 2, lines 1-2 inherently show push circuitry exists to push information specifying a return address down onto the stack).

c) a comparing circuit making a comparison between information specifying a branch destination address of an instruction which can possibly be an instruction equivalent to a subroutine return and the information specifying the return address stored in a top entry of said stack circuit, and outputting a result of a comparison, when the instruction which can possibly be the instruction equivalent to the subroutine return is detected. As explained herein above, this is inherent.

d) an identifying circuit identifying an instruction equivalent to a subroutine return, which corresponds to the instruction equivalent to the subroutine call, based on the result of the comparison (as explained herein above, this is inherent) and,

said identifying circuit identifies the instruction which can possibly be the instruction equivalent to the subroutine return as the instruction equivalent to the subroutine return regardless of the result of the comparison, if the register number of the branch destination address register corresponds to a particular register. Losq has taught that if a potential equivalent to a subroutine return corresponds to a particular register commonly designated as the branch destination address register, (see page 2, lines 2-8. Furthermore, it is notoriously well known in the art to designate a common register. By setting a common register, a usage convention is standardized that makes it easier to maintain compatibility in further developments of a system),

and wherein the predicted branch is the branch of the instruction equivalent to the subroutine return in an architecture for which a particular instruction for a subroutine is not prepared (see page 2, lines 22-32).

10. Referring to claim 7, Losq has taught

a) a stack circuit storing information specifying a return address of a subroutine (See page 1, lines 25-27 and page 2, lines 1-4).

b) a push circuit pushing the information specifying the return address onto said stack circuit, when an instruction equivalent to a subroutine call is detected (Page 1, lines 26-27 and page 2, lines 1-2 inherently show push circuitry exists to push information specifying a return address down onto the stack).

c) a comparing circuit making a comparison between information specifying a branch destination address of an instruction which can possibly be an instruction equivalent to a subroutine return and the information specifying the return address stored in a top entry of said stack circuit, and outputting a result of a comparison, when the instruction which can possibly be the instruction equivalent to the subroutine return is detected. As explained herein above, this is inherent.

d) an identifying circuit identifying an instruction equivalent to a subroutine return, which corresponds to the instruction equivalent to the subroutine call, based on the result of the comparison (as explained herein above, this is inherent) and,

wherein said push circuit does not push the register number of the link register onto said stack circuit if the register number of the link register corresponds to a particular register commonly designated as the branch destination address register, (if the R2 field of BALR instruction corresponds to a particular register, 0, nothing is pushed onto the stack (ESA/390 Principles of Operation, page 7-15, Programming notes: 2.). It is well known in the art to establish a specific register as a usage convention to maintain compatibility when implemented in various systems) and

wherein the predicted branch is the branch of the instruction equivalent to the subroutine return in an architecture for which a particular instruction for a subroutine is not prepared (see page 2, lines 22-32).

11. Referring to claim 8, Losq has taught a pop circuit popping said stack circuit when said identifying circuit identifies the instruction which can possibly be the instruction equivalent to the subroutine return as the instruction equivalent to the subroutine return, and a branch by the instruction equivalent to the subroutine return is taken (see page 2, lines 44-46 show that pop circuitry in the system of Losq is inherent).

12. Referring to claim 9, Losq has taught:

a) a predicting circuit storing branch history information for a branch prediction (Losq's system contains a branch history table in it's predicting circuitry to store branch history information), wherein

b) said comparing circuit makes the comparison between the information specifying the branch destination address and the information specifying the return address, when the branch history information is registered to said predicting circuit. A comparing circuit is inherent as explained herein above including when the branch history information is registered in the branch history table to said predicting circuit.

13. Referring to claim 10, Losq has taught

a) a storing circuit storing information specifying a return address of a subroutine when an instruction equivalent to a subroutine call is detected. On page 1, lines 26-27 and page 2, lines 1-8, Losq has taught a storing circuit consisting of a Branch History Table and Subroutine Return Address Stack storing information specifying a return address of a subroutine when a call is detected.

b) a comparing circuit making a comparison between information specifying a branch destination address of an instruction which can possibly be an instruction equivalent to a subroutine return

and the information specifying the return address stored in said storing circuit, and outputting a result of the comparison, when the instruction which can possibly be the instruction equivalent to the subroutine return is detected. On page 2, lines 40-46, Losq has taught a decoder used as comparing circuitry to compare R1 of a BALR or similar instruction on the return stack, which holds information specifying the return address, and R2 of a BCR or similar instruction which holds the branch destination address of a potential subroutine return instruction.

c) an identifying circuit identifying an instruction equivalent to a subroutine return, which corresponds to the instruction equivalent to the subroutine call, based on the result of the comparison (it is inherent in the system disclosed by Losq, that identifying circuitry must be present since Losq's system is able to identify an instruction equivalent to a subroutine return) and,

a circuit invalidating the information stored in said storing circuit when an event which causes correspondence between a subroutine call and a subroutine return to be improper. A circuit invalidating information stored in said storing circuit is inherent since the return address stack is not popped when there is a improper correspondence between a subroutine call and return (see page 2, lines 44-47 and page 3, lines 1-22), and

wherein the predicted branch is the branch of the instruction equivalent to the subroutine return in an architecture for which a particular instruction for a subroutine is not prepared (see page 2, lines 22-32).

14. Referring to claim 11, Losq has taught:

a) a predicting circuit storing branch history information for a branch prediction (as taught herein above);

b) a setting circuit setting in said predicting circuit a flag indicating that a return destination of a detected instruction equivalent to a subroutine return differs, when an instruction equivalent to a subroutine return, which does not return to an instruction address immediately succeeding the instruction equivalent to the subroutine call, is detected. It is inherent the predicting circuit of Losq must use some type of flag to indicate that a return destination of potential subroutine return differs or the system would not function correctly.

15. Referring to claim 12, it is inherent the system disclosed by Losq has said predicting circuit comprises a return address stack circuit storing the return address of the subroutine, pops said return address stack circuit if the flag is recognized at the time of a branch prediction, and does not use a popped return address as a predicted branch destination. It is necessary to be able to pop a return address on a return address stack if a return address has to be removed but not used.

16. Referring to claim 13, Losq has taught:

a) a storing circuit storing information specifying a return address of a subroutine when an instruction equivalent to a subroutine call is detected. On page 1, lines 26-27 and page 2, lines 1-8, Losq has taught a storing circuit consisting of a Branch History Table and Subroutine Return Address Stack storing information specifying a return address of a subroutine when a call is detected.

b) a comparing circuit making a comparison between information specifying a branch destination address of an instruction which can possibly be an instruction equivalent to a subroutine return and the information specifying the return address stored in said storing circuit, and outputting a result of the comparison, when the instruction which can possibly be the instruction equivalent to

the subroutine return is detected. On page 2, lines 40-46, Losq has taught a decoder used as comparing circuitry to compare R1 of a BALR or similar instruction on the return stack, which holds information specifying the return address, and R2 of a BCR or similar instruction which holds the branch destination address of a potential subroutine return instruction.

c) an identifying circuit identifying an instruction equivalent to a subroutine return, which corresponds to the instruction equivalent to the subroutine call, based on the result of the comparison (it is inherent in the system disclosed by Losq, that identifying circuitry must be present since Losq's system is able to identify an instruction equivalent to a subroutine return) and,

a predicting circuit storing branch history information for a branch prediction (as explained herein above);

a circuit performing a control such that a predetermined flag is set when an instruction equivalent to a subroutine call, which is unregistered in the branch history of said predicting circuit, is detected, the predetermined flag is reset when an instruction equivalent to a subroutine return, which corresponds to the unregistered instruction equivalent to the subroutine call, is detected, and the instruction equivalent to the subroutine return corresponding to the unregistered instruction is not identified as an instruction equivalent to a subroutine return is said predicting circuit,

the predicting circuit predicting the branch responsive to the identifying and the control (It is inherent the system disclosed by Losq must be able to handle instruction equivalent to a subroutine calls not registered in the branch history of said predicting circuit. Initially, the BHT

will be empty and the system of Losq must have the capabilities to handle instruction equivalent subroutine calls that are not yet registered), and

wherein the predicted branch is the branch of the instruction equivalent to the subroutine return in an architecture for which a particular instruction for a subroutine is not prepared (see page 2, lines 22-32).

17. Referring to claims 14 and 18, Losq has taught

a) a return address stack circuit storing a return address of a subroutine when an instruction equivalent to a subroutine call is detected (see page 1, lines 24-26);

b) a comparing circuit making a comparison between a branch destination address of an instruction which can possibly be an instruction equivalent to a subroutine return, and the return address stored in said return address stack circuit, and outputting a result of the comparison, when the instruction which can possibly be the instruction equivalent to the subroutine return is detected. This is inherent as explained herein above.

c) an identifying circuit identifying an instruction equivalent to a subroutine return, which corresponds to the instruction equivalent to the subroutine call, based on the result of the comparison. This is inherent as explained herein above.

d) wherein the predicted branch is the branch of the instruction equivalent to the subroutine return in an architecture for which a particular instruction for a subroutine is not prepared, and wherein only the taken instruction of a branch instruction is thus registered (see page 2, lines 22-32).

18. Referring to claim 15, the method in claim 15 does not recite limitations above the claimed device set forth in the above claims and is therefore rejected for the same reasons set forth in the rejection of the above claims.

19. Referring to new claim 19, the limitations of the claim are rejected on the same basis as similar limitations as seen supra.

Response to Arguments

20. Applicant's arguments filed 1/30/04 have been fully considered but they are not persuasive.

21. On page 12, Applicant argues referring to claim 6 in essence that:

"Losq does not address a commonly designated as the branch destination address register"

As taught in the rejection above, it is well known in the art to establish a specific register as a usage convention to maintain compatibility when implemented in various systems.

22. On page 12, Applicant argues referring to claim 7 in essence that:

"Nothing is said about not pushing when the register is the commonly used branch destination address register."

As taught in the rejection above, it is well known in the art to establish a specific register as a usage convention to maintain compatibility when implemented in various systems. Furthermore, nothing is pushed onto the stack when the register is zero as taught above.

23. On page 12, Applicant argues referring to claim 10 in essence that:

"Losq does not address invalidating contents of the storing circuit when a non-correspondence event occurs, such as an interrupt."

Losq does teach this as seen on page 3 and taught in the rejection above.

24. On page 12, Applicant argues referring to claim 13 in essence that:

"Claim 13 recites features relating to an ability to perform branch prediction when the instruction equivalent to a call instruction is not registered in the branch history and Losq does not teach this."

The rejection of claim 13 above has been clarified to show that Losq must teach this.

25. On page 13, Applicant has arguments referring to claim 1, however, claim 1 was cancelled.

26. In regards to the arguments in respect to claims 4 and 14-18, see the 112 rejection above.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Justin Knapp whose telephone number is (703) 308-6132. The examiner can normally be reached on Mon - Fri 9 am - 5:30 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Jeffrey Gaffin can be reached on (703) 308-3301. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

JEFFREY GAFFIN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

Justin Knapp
Examiner
Art Unit 2182